COSC 4330/6310 FINAL EXAMINATION MAY 4, 2012

CLOSED BOOK. YOU CAN HAVE ONE PAGE OF NOTES. CHEATERS RISK EXPULSION FROM THE UNIVERSITY.

1. Answer in one or two sentences to the following questions (4×5 points).

a) What is the major disadvantage of using ***very large blocks*** in a file system?

Using very large blocks will result in excessive internal fragmentation whenever the file system contains many small files.

b) What is the major advantage of the ***Windows*** page replacement policy?

The windows page replacement policy supports real-time processes because it can allocate to a given process enough pages to guarantee there will be no page faults during its execution.

c) How can you prevent deadlocks by denying the ***circular wait condition***?

We can prevent deadlocks by forcing all processes to acquire their resources in the same linear order.

d) Where do UNIX file systems store their ***access control lists***?

UNIX stores file access control lists in the i-node of each file.

---

2. Consider a file whose access control bits are **rw-r-----**. Which system users can

a) Execute the file? (5 points) None.

b) Read the file? (5 points) The owner of the file and the users specified in the file group.

3. A Berkeley UNIX file system has 32-bit addresses, 15 block addresses in each i-node and 8 kilobyte blocks. How many ***bytes*** can be accessed:

a) directly from the i-node: $12 \times 8KB = 96\ KB$
(5 points)

b) with one level of indirection: $(8K/4) \times 8KB = 16\ MB$
(5 points)

c) with two levels of indirection: $4\ GB - 16\ MB - 96\ KB$
(5 points)

4. Compare and contrast access control lists and tickets (2×5 points)

a) ***Main advantage of access control lists:***

They are more flexible: we can and delete authorized users as well as changing their privileges at any time.

b) ***Main advantage of tickets:***

They are faster than access control lists: we do not need to verify the identity of ticket holders.

---

5. A 64-bit computer system has 128 GB of main memory and 4KB pages.

a) How many page frames are there in the main memory? (5 points)

$128G/4K = 32\ M$ (or $2^{37}/2^{12} = 2^{25}$) **frames**

b) Which page table organization would allow all page tables to fit into main memory? (5 points)

An inverted page table (it only has entries for pages that are in main memory).

c) Assuming that each page table entry occupies 24 bytes, which fraction of the main memory would be occupied by the page tables? (5 points) (***Hints:*** *express your answer as a fraction and do not consider the case of shared pages.*)

24 bytes per page frame = $3 \times 2^3/2^{12} = 3/2^9$ (or $6/2^{10} \approx 0.6\%$)

6. Which of the following statements are ***true*** or ***false*** (2 points) and ***why***? (3 points)

a) The ***missing bit*** indicates which pages must be *written back to disk* when they are expelled.

FALSE, it indicates which pages are not in main memory.

b) It is easier to eliminate a *circular wait* condition than a *hold and wait* condition.

TRUE, eliminating a circular wait condition requires all resources to be allocated at the same time, which is more constraining than imposing a fixed linear order on all resource requests.

c) The LRU policy always expels the page that has been in memory for the longest period of time.

FALSE, it expels the page that has not been ACCESSED for the longest period of time.

d) A good page fault rate for a virtual memory system is one page fault every one thousand to two thousand references.

FALSE, it is much too high.

e) When we create a new file, we should always create the new i-node before creating the directory entry that points to it.

TRUE, otherwise a crash could leave the file system in a state where the new directory entry could contain the address on an i-node that does not yet exist.

f) When we delete a file, we should always delete the old i-node before deleting the directory entry that points to it.

FALSE, a crash could leave the file system in a state where the old directory entry could contain the address on an i-node that was already deleted.